2020-07-21

Composing Popular Music with Physarum polycephalum-based Memristors

Venkatesh, S

http://hdl.handle.net/10026.1/16543

Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)
Birmingham City University

Composing Popular Music with Physarum polycephalum-based Memristors

Satvik Venkatesh
Interdisciplinary Centre for Computer Music Research
University of Plymouth
United Kingdom
satvik.venkatesh@plymouth.ac.uk

Edward Braund
Interdisciplinary Centre for Computer Music Research
University of Plymouth
United Kingdom
edward.braund@plymouth.ac.uk

Eduardo Reck Miranda
Interdisciplinary Centre for Computer Music Research
University of Plymouth
United Kingdom
eduardo.miranda@plymouth.ac.uk

ABSTRACT

Creative systems such as algorithmic composers often use Artificial Intelligence models like Markov chains, Neural Networks, and Genetic Algorithms to model stochastic processes. Unconventional Computing (UC) technologies explore non-digital ways of data storage, processing, input, and output. UC paradigms such as Biocomputing and Quantum Computing delve into domains beyond the binary bit to handle complex non-linear functions. In this paper, we harness *Physarum polycephalum* as a memristor to process and generate creative data for popular music. The organism works as a collaborator in the process of composing our song titled *Creep into my Lawn.* While there has been research conducted in this area, the literature lacks examples of popular music and how the organism's non-linear behaviour can be controlled while composing music. This is important because non-linear forms of representation are not as obvious as conventional digital means. This study aims at disseminating this technology to non-experts and musicians so that they can incorporate it in their creative processes. Furthermore, it combines resistors and memristors to have more flexibility while generating music and optimises parameters for faster processing and performance.

Author Keywords

Unconventional Computing, Biocomputing, Musical Composition, Popular Music, Physarum polycephalum

CCS Concepts

•Theory of computation → Unconventional Computing; •Applied computing → Sound and Music Computing; Algorithmic Composition;

1. INTRODUCTION

There is a vast corpus of research that explores computational processes to exhibit creative behaviour [8]. The process of musical composition frequently involves two elements — material that has been previously heard and ideas that are new and innovative [9]. The former encompasses characteristics of the composer's style and their inspirations. The latter has been explored in computational creativity through stochastic processes, which aim to induce randomness in the output of the system [19].

Many Artificial Intelligence (AI) models such as Markov chains [12], Genetic Algorithms (GA), Artificial Neural Networks (ANNs) [23] have been successfully adopted to render creative systems. Markov chains compute a conditional probability distribution of future states depending on the current state [19]. The architecture of ANNs is inspired by the structure of the brain and enables us to model complex non-linear functions. Additionally, other models such as self-similarity and cellular automata [14], that do not strictly fall under the AI bracket, have been implemented to render creative systems [10].

Physical, chemical, and biological processes have not only inspired AI algorithms, but given rise to other forms of computation. Unconventional Computing (UC) aims to develop new computer architectures for data storage and processing [1]. Computational creativity has been conventionally rendered on classical computers based on the von Neumann architecture. The fundamental unit of processing and storage is the binary bit, that is either 0 or 1. Conversely, UC technologies explore analog forms of computation and storage. For example, quantum computing uses *qubits* as the fundamental unit of processing.

Physarum polycephalum is a plasmodial slime mould that has been widely harnessed for biocomputing applications. Researchers have utilised its ability to optimise its shape according to the environment to solve mazes [16], build logic gates [2], and find shortest paths [17]. Another interesting property of the organism is that it can act as a memristor [11]. Memristor (a portmanteau of *memory* and *resistor*) is the fourth fundamental passive circuit element, the first three being resistor, capacitor, and inductor. It was theoretically demonstrated by Leon Chua in 1971 [7], but was physically realised in 2008 [22].

Research into memristor technology is gaining traction due to many reasons. It combines storage and processing into one unit and therefore, has the potential to optimise computing architectures. [21, 13] demonstrated spike-timing-dependent plasticity (STDP)-like behaviour using memristors. STDP has been adopted in feed-forward ANNs for pattern recognition algorithms [3].

Physarum polycephalum-based memristors serve as an inexpensive and accessible UC prototype, when compared to quantum computers and chemically manufactured memristors. [5] designed 3D printed receptacles to enclose the organism into a portable electrical component. [6] presented a compact kernel to interface with biomemristors for creative practitioners and D.I.Y. prototyping enthusiasts. Moreover, research is progressing with regards to making UC technologies available to individuals outside laboratories. This

NIME'20, July 21-25, 2020, Royal Birmingham Conservatoire, Birmingham City University, Birmingham, United Kingdom.

makes it increasingly important to use these bespoke systems in creative works.

This paper explores how *Physarum polycephalum*-based memristors can be used in the process of composing popular music. It works with *Physarum polycephalum* as a collaborator and is a step towards the long-term dream of building a rock band of biomemristors. As we are using a non-linear component as the processing device, input parameters that govern the composition process require non-digital forms of representation. This study renders a practice-based approach towards understanding non-linear input parameters, which may not be as obvious as binary logic. It aims to break down the creative process and explores how these parameters influence the output of the PhyBox. *Biocomputer Rhythms* by Eduardo Miranda was the first musical piece for piano and percussion composed with the *PhyBox* [15]. However, it was not composed in a popular music context and there have been recent developments with mapping procedures, such as resistors can be used alongside memristors in the process of composition [24]. This paper incorporates these developments, optimises parameters to minimise processing time, and offers more degree of freedom to the user.

2. SYSTEM OVERVIEW

2.1 Hardware-wetware Framework

We maintain a *Physarum polycephalum* farm [1] grown in plastic lunch boxes as shown in figure 1. In order to grow *Physarum polycephalum*-based memristors and make them a part of the composition process, we transfer a section of the organism into receptacles [5]. These receptacles allow the organism to act as an electrical component and are operated through the PhyBox [6], a portable kernel for creative applications as shown in figure 2. PhyBox is powered by a Raspberry Pi model B.

Figure 1: *Physarum polycephalum* culture grown in a lunch box.

Training data of music is converted into voltages with the help of a mapping procedure [24]. In this paper, training data refers to the music originally composed by the author. These voltages are serially applied across the memristors as shown in figure 3. The current flowing through the memristor depends on its resistance (or memristance). The current is again translated to music through a similar mapping procedure. Therefore, the memristors create new music based on the training data.

Music is fed into the PhyBox through the MIDI protocol. MIDI data is broken down into four parameters — pitch, velocity, time between note-ons (rhythm), and duration. Each memristor processes one of these parameters in parallel, that is there are 4 memristors in the circuit.

Figure 2: Setting up the PhyBox for musical composition.

2.2 Mapping procedure

The framework for generating creative data harnesses the spiking behaviour of memristors. Each time there is a voltage change across the memristor, there is a spike in current as shown in figure 3. This behaviour is different from that of linear components like resistors in electronic circuits. The following explains its behaviour. When the positive change in voltage is produced across the memristor, the resistance is initially low. With time, it develops resistance and there is a gradual decrease in the current flowing through the circuit.

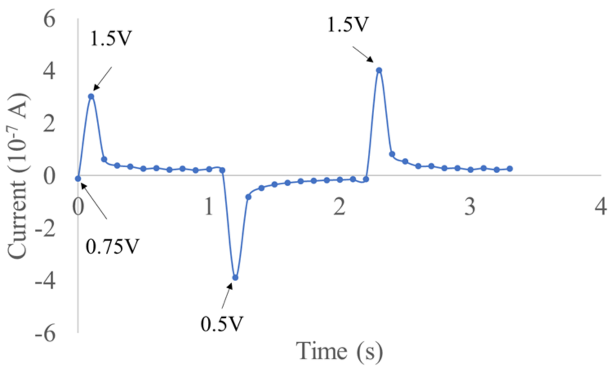

Figure 3: Spike generated by the memristor on applying four voltage changes — 0.75V, 1.5V, 0.5V, and 1.5V. Positive spikes are produced for positive voltage changes and vice versa.

2.2.1 *Music to Voltages*

In this system, these voltage transitions are attributed to musical note transitions. A musical note is assigned to a specific voltage on a *priority* basis. Musical notes that occur more often are assigned higher priority and vice versa. If there are two events with the same number of occurrences, the more recent event is given a higher priority. This is analogous to the learning rule in STDP, where the new event is good and the old event is bad.

As we are using a Raspberry Pi, the voltage range is 0 to 3V. It is assigned by equation 1 [24].

$$v_{\mathrm{a}} = \frac{3}{N}(p - \frac{1}{2})Volts \quad (1)$$

where v_a is the assigned voltage, N is the number of musical notes and p is the priority of the event. A relationship between popularity and assigned voltages is devised by the following analogy. The rate of change of current (or slope) in the beginning of the graph is greater when compared to the end. Moreover, taller spikes are generated for greater changes in voltage. The stability of resistance is related to the popularity of the event. Therefore, more popular events are mapped to lower voltages and less popular events are mapped to higher voltages.

Note that when the training data is initially fed into the PhyBox, there are no events in the memory. It is trained during the course of the song. Therefore, the number of variations that can possibly occur gradually increases with more data.

2.2.2 *Current to Music*

A similar procedure is followed while converting current readings to music. The voltage range is well-defined to be 0 to 3V. However, a discrete current range is not defined because the resistance of the memristor is unknown. Empirically, studies have approximated the resistance in the order of mega-ohms [6]. Let $I_{\max}$ and $I_{\min}$ be the maximum and minimum current sensed during the spike respectively. The current range $[I_1, I_2]$ is defined by the following equation.

$$[I_1, I_2] = \Big[\frac{min(V_{\mathrm{a}})}{(v_{\mathrm{a}}/I_{\min})}, \frac{max(V_{\mathrm{a}})}{(v_{\mathrm{a}}/I_{\max})}\Big] \quad (2)$$

where V_a is the set of all assigned voltages and v_a the input voltage at that specific moment in time. The current range is divided into equal divisions and converted to music based on *midrise quantisation* [4]. Again, the events are ordered according to *priority*, where lower current values stand for more popular events and vice versa.

2.2.3 *Resistors*

One of the advantages of this mapping procedure is that it provides a way to incorporate resistors in the process of musical composition. It defines a resistor's behaviour as linear. Therefore, when the training data is fed into a resistor, the output of the system is exactly identical to the input. This offers creative pathways to explore in the process of composition, such as the rhythm can be identical to the input and the pitch can be composed by the memristor. It is important to note that the value of the resistance does not influence the output. In this study, we used $330k\Omega$ resistors.

2.3 Parameters

There are two parameters that monitor the output produced by the system — dwell time and measurement offset. Dwell time specifies how long the voltage is applied across the memristor. This basically determines the processing time of the memristors for each musical event. For example, if the dwell time is 1 s, the memristor takes 20 s to process 20 musical events. The measurement offset determines at what point of time is the current value calculated in the circuit. For instance, on applying 1.5 V at 0.1 s in figure 3, 0% offset considers the current at 0.1 s and 100% offset considers the current at 1.1 s (Here the dwell time was 1 s). [6, 24] have conducted some experiments to understand how these non-linear parameters influence the output of the memristors. 0% offset produces smaller deviations from the input data when compared to 100% offset [24]. However, there is still not a clear delineation of these functions due to their complex non-linear behaviour.

3. EXPERIMENTS

In this paper, we use *Physarum polycephalum* as a collaborator in the process of musical composition. As the current interface only accepts MIDI data, we have harnessed it to compose parts for piano, drums, and the vocal melody. The dwell time was constant at 1 s. The measurement offset was altered between 0%, 30%, 50%, and 100%.

3.1 Structure

The time and key signature of the song is 4/4 and E minor respectively. The genre of the song falls under Rock. The song starts with the main riff played only on the piano, which converges into the other instruments joining the main riff. Followed by that, we have one verse, pre-chorus, and chorus, which are sung by a vocalist. The chorus has the main riff in the background. After the chorus, there is a piano solo followed by a drum solo. The song finishes with another chorus at the end. This song follows the structure adopted by most popular music compositions. However, it was slightly altered, for instance omitting the second verse, in order to make things more interesting in the view of *Physarum polycephalum*.

In order to explain the structure of the song with clarity, the verse was divided into two parts. While feeding in data for piano and drums, we divided the whole song into 3 blocks. This was done to avoid overloading the model with too many possibilities and transitions. The first block consisted of the intro and first part of the verse. The second block comprised the second part of the verse and the pre-chorus. The third block contained the remainder of the song. These blocks were not divided with regards to time, but were divided with respect to number of MIDI events and structure of the song. For instance, chords in the verse contained more MIDI events than melody lines. The vocal melody was transcribed as a MIDI file. MIDI events like sustain pedal was removed from the training data.

3.2 Main Riff

The main riff of the song is 4 bars in length. A piano tune is backed up by drums and guitars for the main riff. The first two bars of the piano melody were composed by the author. We wanted the third and the fourth bar to follow a similar pattern as the first two, but with minor variations. The aim is to create a feeling of repetition alongside some anticipation in the listener. We conducted experiments with measurement offset values of 0%, 30%, and 100%. The guitars and drums were composed by keeping the piano track as a reference.

3.3 Verse

The instrumentation for the verse consisted of vocals, piano, and drums. The first part of the verse had simple piano chords at the beginning of each bar. We experimented with measurement offset values of 0%, 30%, and 100%. The objective of this experiment was to make *Physarum polycephalum* improve the chord pattern, chord inversions, and the voicing.

The second part of the verse had arpeggios played on the piano. For this section of the song, we felt that there was less flexibility with altering the rhythm. We wanted to maintain the rhythmic structure originally composed by the author. Therefore, we attached memristors for pitch and velocity and resistors for time between note-ons and duration. The measurement offset was set to 100% and this configuration was referred to as 100R.

3.4 Piano Solo

The harmonic and rhythmic complexity of the piano solo is higher than the other sections of the song. The piano solo was originally composed by the author. Also, there was a scale shift within the solo from E minor to E Phrygian and then C major. At the end of the piano solo, the song returns to E minor. We wanted to explore how *Physarum polycephalum* would respond to scale changes and complex melodic ideas. This is the most interesting section of the song because it provides a high degree of freedom with regards to musical options. Additionally, it does not restrict *Physarum polycephalum* with respect to the song's instrumentation because the other instruments do not play during the piano solo. Tests were conducted for measurement offsets of 5%, 30%, 50%, 100%, and 100%R.

3.5 Drums

We wanted to harness the organism to create new drum fills based on existing ones. Additionally, we wanted to explore its creativity for the drum solo. We tested it with measurement offset values of 0%, 30%, 100%, and 100%R.

3.6 Vocal Tune

A vocal melody was composed by the author and transcribed as a MIDI track. Our objective of using *Physarum polycephalum* for the vocal tune was to make it sound more catchy and less monotonous. We explored measurement offsets of 0%, 30%, 100%, and 100%R.

3.7 Dissimilarity

In order to objectively calculate the difference between original melody and *Physarum polycephalum*'s rendition, we defined a metric called *dissimilarity*, which is the number of musical events that are different from the original input. Each of the parameters — pitch, velocity, time between note-ons, and duration would return a value of dissimilarity. The *total dissimilarity* is defined as the sum of individual dissimilarities of the four parameters.

4. RESULTS

All audio and MIDI files that were composed by the author and *Physarum polycephalum* can be found in this link[1]. The naming format for the files is {block _Instrument _Type _Offset}. The title of the song is *Creep into my Lawn*. A music video that illustrates the parts composed by Physarum polycephalum and the author is available in this link[2]. The model used to generate musical data only uses note transitions. This gives high flexibility to *Physarum polycephalum*, but opens up possibilities like time signature mismatch and disharmonious chords. Therefore, few modifications were made by the composer to suit the song. As the model is based on musical transitions, in *Physarum polycephalum*'s output, the first note will be missing because no transition has occurred yet. Hence, while evaluating its output, the first note was added from the input.

4.1 Dissimilarity

Table 1 contains the *total dissimilarities* for different values of measurement offset. Consistently, 0% was more similar to the input when compared to other offsets. 30% was closer to the input when compared to 100%, but the difference was much smaller.

4.2 Main Riff

The main riff originally composed by the author is shown in figure 4. *Physarum polycephalum*'s output for measurement offset of 0% was not interesting. It resembled the original melody except for changing one crotchet to a dotted crotchet. For measurement offsets of 30% and 100%, the generated output sounded more appropriate for the main riff. The authors felt that 30% offset had more resemblance to the original tune when compared to 100%. We chose 100% and made no modifications to *Physarum polycephalum*'s output. It is shown in figure 5. For the third and fourth bar of the main riff, guitar and drums parts were re-composed to match the new tune composed by the memristors.

Figure 4: The original main riff composed by the author.

Figure 5: *Physarum polycephalum*'s rendition of the main riff for measurement offset of 100%.

4.3 Verse Part 1

The first part of the verse essentially contained piano chords, drums, and vocals. 0% offset subtly altered the harmonic structure of the verse. 30% output collapsed all the individual notes into one point. This was because the time between the chords was 0, which was obviously not useful for the composition. 100% made too many variations that did not suit the genre of the song and it was difficult to perceive the harmonic structure. Hence, we chose 0% offset for this section.

The original chord pattern of the verse was *Em—G—Am—D*. All chords were first inversions. While processing the output of *Physarum polycephalum*, the root notes of chords were left unchanged. The first chord was the same as the input. The second chord was a triad of the notes $E4$, $G4$, and $A4$ (the numbers stand for the octave), which sounded very close to the chord *Esus4/G*. In order to form a triad that sounds harmonically more stable, $G4$ was changed to $C5$ and thus, obtaining *Esus4/G*. The third chord generated by *Physarum polycephalum* contained $E2$ as the lowest note, which sounded like the second inversion of *Am*. Therefore, the third chord was changed to the second inversion of *Am*. The fourth chord generated by *Physarum polycephalum* contained $F\#4$, $G4$, and $A4$, which sounded close to the chord *Gsus2/D*. However, $F\#4$ and $G4$ are only a semi-tone apart and therefore, made the chord sound unstable. Hence, $F\#4$ was modified to $D5$ and thus obtaining the chord *Gsus2/D*.

The original vocal tune composed by the author was a simple melody line. 0% produced a melody that was similar to the input, but had irregular rhythms and few variations in pitches. It did not fulfil our goals of making the vocal tune sound more catchy. 30% produced interesting variations in the output. However, the output sounded very continuous to suit a vocal line. Its rhythmic structure did not have enough stops. 100% offset formed a good foundation for the vocal line. Furthermore, it went well with the new chords

[1]https://github.com/satvik-venkatesh/Physarum-Music
[2]https://youtu.be/dKQhnlO7ZrI

Table 1: Table of tests conduced with PhyBox for different values of measurement offset (MO). Numbers within the MO columns show the *total dissimilarity* and '-' indicates no test was conducted. The 'final song' column shows the values of MO chosen for the final song. The values in bold indicate that they were modified to suit the song. There are repetitions of values across rows because the dissimilarities are calculated for the whole block and not only the section.

Section	Physarum MO (%)					Final Song
	0%	30%	50%	100%	100%R	
Main Riff	69	214	-	261	-	100%
Verse Pt. 1 Piano	69	214	-	261	-	**0%**
Verse Pt. 2 Piano	-	-	-	-	150	**100%R**
Pre-chorus Piano	-	-	-	-	150	**100%R**
Piano Solo	414	738	743	860	313	50% and **100%R**
Verse Pt. 1 and Pt. 2 Vocals	106	229	-	265	81	**100%**
Pre-chorus and Chorus Vocals	106	229	-	265	81	-

composed by *Physarum polycephalum*. However, few notes were omitted and altered in order to suit the song.

4.4 Verse Part 2

The rhythmic structure was unaltered because resistors were connected instead of memristors for time between note-ons and duration. During the second bar, the memristors produced an accidental *Bb* which was not anticipated by the composer. This accidental (or blue note) is pretty popular in jazz and blues genres, which gave the song a new flavour. After performing few modifications, the arpeggio composed by *Physarum polycephalum* was fit into the song.

4.5 Pre-chorus

The pre-chorus of the song mainly consisted of strumming and piano patterns in close association with the drumbeat and hence, we wanted to maintain the rhythmic structure. For this part of the song, we used resistors for rhythmic parameters and memristors for pitch and velocity. The memristors changed the voicing of chords and also performed some chord inversions. Few modifications were made to incorporate it in the song.

For the vocal tune of the pre-chorus, we felt that the output of memristors did not match the backing chords. Furthermore, the original melody line went well with the subtle changes produced by memristors in the piano line. Therefore, we used the original tune.

4.6 Chorus

The chorus of the song had the main riff running in the background alongside vocals. Therefore, the instrumental section was a replica of what was done earlier.

In order to compose the vocal tune for the chorus, we analysed the output of resistors connected to rhythmic parameters. For the chorus of the song, we felt that the original version was more suitable and *Physarum polycephalum*'s output did not improve it. As we progressed towards the chorus, the song was gaining structure and form. Furthermore, it had the main riff of the song playing in the background. Therefore, there was more rigidity in the number of suitable musical options. Hence, we chose the original tune created by the composer.

4.7 Piano Solo

Among all values of measurement offset, 50% was chosen due to the following explanation. In the original solo composed by the author, there was a mode shift from E minor to E Phrygian and then C major. Among all outputs produced by *Physarum polycephalum*, 50% created a solo that was distinctly perceived as G major. Surprisingly, there was no training data that existed in G major. However, it is the relative major of the E minor scale. This was a very interesting observation and *Physarum polycephalum* sounded like a highly skilled pianist! This demonstrated that the intelligence of the organism can be harnessed to create unique trajectories in creative processes. There were no modifications made by the composer on *Physarum polycephalum*'s output. Furthermore, 30% offset created few bars that were perceived as G major. The output of 30% and 50% were similar, but 50% was preferred by the composer. 100% offset created many stops during the solo which made it unsuitable. Additionally, 100% offset sounded distinctly different from 30% and 50%.

The final piano solo starts with the original one composed by the author. This was mutated with outputs produced by 0% and 100%R. Just before the scale shift to C major, *Physarum polycephalum*'s solo in 50% offset begins, which lasts for 9 bars. In the final song, this part spans from 2 min 22 s to 2 min 42 s.

4.8 Drums

For all values of measurement offset, the musical files sounded random and unsuitable for the song. This might be due to the 4 parameters — pitch, velocity, time between note-ons, and duration are not suitable to analyse percussion tracks. It might need higher-level musical structures such as time signatures. Furthermore, in such a song, drums are generally a supporting instrument. *Physarum polycephalum*'s output sounded more like a melody line and not a drumbeat. Due to the above observations, we decided to not use *Physarum polycephalum* for the drum solo.

4.9 Discussion

This paper presented the process of composing with *Physarum polycephalum*-based memristors. As mentioned earlier, the model adopted for composition uses note transitions. Therefore, it does not account for higher-level musical structures like time signatures and harmonic functions. Therefore, we had to make modifications in *Physarum polycephalum*'s output to incorporate it in the song. It is important to note that modifications to the output produced by *Physarum polycephalum* was only done in parts where there was a high involvement of multiple tracks and presence of chords. For instance, slightly altering piano and vocal lines during the verse, but not making any modifications in the main riff. Furthermore, individual tracks were passed separately into the PhyBox, which may have contributed to this behaviour.

As no changes were made in outputs produced for the main riff and piano solo, it suggests that the model can be improved by further understanding higher-level structures. Examining how instruments interact with each other opens up possibilities of creating networks of memristors that collectively produce a coherent and ready-to-use output.

The author felt that the memristors have changed the original composition significantly and added its own footprint. It was an interactive process where *Physarum polycephalum* makes suggestions and the composer evaluates them. However, it is important to note that many observations and results were drawn through subjective measures and aesthetic preferences of the composer.

This paper used different values of measurement offset for different sections of the song. In figure 3, it can be observed that 0% captures the value at the top of the spike. The next sample recorded by the PhyBox is after a steep descent. Therefore, in order to have more flexibility in measurement offset, the sampling rate of the system needs to be increased.

We used *dissimilarity* as the metric to compare different values of measurement offsets. This can be improved by borrowing concepts like *entropy* from information theory [20, 18]. This will enable us to explore the statistical structure of music in greater detail.

5. CONCLUSION

This paper presented a composition that was in collaboration with *Physarum polycephalum*. It showed how the non-linear behaviour of memristors can be harnessed for the process of popular music production. It aims to make UC technologies reach a wider audience so that non-experts in UC and musicians can adopt them in the process of music production. Harnessing *Physarum polycephalum* has great potential in computational creativity and popular music.

We showed that resistors can be creatively incorporated alongside memristors during musical composition. We demonstrated that a dwell time of 1 s is suitable for obtaining the memristor's non-linear behaviour. This is faster than an earlier model that used 2 s [6]. *Dissimilarity* values show that the measurement offset considerably influenced the melody produced by the memristor. For each parameter, the output created by *Physarum polycephalum* was unique. This can be verified with sound files provided in the database.

In this paper, we trained the model with only one song. Future work would explore how the system can be scaled up. This might include creating more complicated learning models such as networks of memristors. Additionally, higher-level data structures such as time signatures and harmonic functions need to be explored.

6. ACKNOWLEDGEMENTS

The authors thank Dr. David Moffat for proofreading the article and offering constructive feedback.

7. REFERENCES

[1] A. Adamatzky. *Physarum Machines: Computers from Slime Mould*, volume 74 of *A*. World Scientific, 2010.

[2] A. Adamatzky, J. Jones, R. Mayne, S. Tsuda, and J. Whiting. Logical gates and circuits implemented in slime mould. In *Advances in Physarum Machines*, pages 37–74. Springer, 2016.

[3] O. Bichler, D. Querlioz, S. J. Thorpe, J.-P. Bourgoin, and C. Gamrat. Extraction of temporally correlated features from dynamic vision sensors with spike-timing-dependent plasticity. *Neural Networks*, 32:339–348, 2012.

[4] M. Bosi and R. E. Goldberg. *Introduction to Digital Audio Coding and Standards*. Kluwer Academic Publishers, 2003.

[5] E. Braund and E. R. Miranda. On building practical biocomputers for real-world applications: Receptacles for culturing slime mould memristors and component standardisation. *Journal of Bionic Engineering*, 14(1):151–162, 2017.

[6] E. Braund, S. Venkatesh, and E. R. Miranda. Phybox: A programmable interface for physarum polycephalum-based memristors. *International Journal of Unconventional Computing*, 14(3-4):217–233, 2019.

[7] L. Chua. Memristor-the missing circuit element. *IEEE Transactions on circuit theory*, 18(5):507–519, 1971.

[8] S. Colton, G. A. Wiggins, et al. Computational creativity: The final frontier? In *Ecai*, volume 12, pages 21–26. Montpelier, 2012.

[9] D. Cope. *Experiments in Musical Intelligence*. A-R Editions, Inc, 1996.

[10] J. D. Fernández and F. Vico. Ai methods in algorithmic composition: A comprehensive survey. *Journal of Artificial Intelligence Research*, 48:513–582, 2013.

[11] E. Gale, A. Adamatzky, and B. de Lacy Costello. Slime mould memristors. *BioNanoScience*, 5(1):1–8, 2015.

[12] K. Jones. Compositional applications of stochastic processes. *Computer Music Journal*, 5(2):45–61, 1981.

[13] B. Linares-Barranco and T. Serrano-Gotarredona. Memristance can explain spike-time-dependent-plasticity in neural synapses. *Nature Precedings*, 4, 09 2010.

[14] E. R. Miranda and J. Al Biles. *Evolutionary computer music*. Springer, 2007.

[15] E. R. Miranda, E. Braund, and S. Venkatesh. Composing with biomemristors: Is biocomputing the new technology of computer music? *Computer Music Journal*, 42(3):28–46, 2018.

[16] T. Nakagaki, H. Yamada, and Á. Tóth. Intelligence: Maze-solving by an amoeboid organism. *Nature*, 407(6803):470, 2000.

[17] T. Nakagaki, H. Yamada, and A. Toth. Path finding by tube morphogenesis in an amoeboid organism. *Biophysical chemistry*, 92(1-2):47–52, 2001.

[18] M. T. Pearce and G. A. Wiggins. Auditory expectation: the information dynamics of music perception and cognition. *Topics in cognitive science*, 4(4):625–652, 2012.

[19] S. M. Ross. *Stochastic processes*. JohnWiley & Sons, Inc, 2 edition, 1996.

[20] C. E. Shannon. A mathematical theory of communication. *Bell system technical journal*, 27(3):379–423, 1948.

[21] G. S. Snider. Spike-timing-dependent learning in memristive nanodevices. In *Proceedings of the 2008 IEEE International Symposium on Nanoscale Architectures*, pages 85–92. IEEE Computer Society, 2008.

[22] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams. The missing memristor found. *nature*, 453(7191):80, 2008.

[23] P. M. Todd. *Computer Music Journal*, 13(4):27–43, 1989.

[24] S. Venkatesh, E. Braund, and E. R. Miranda. A nonlinear approach to generate creative data using physarum polycephalum-based memristors. In A. Adamatzky, editor, *Slime Mould in Arts and Architecture*. River Publishers, 2019.